



Article

Why Is Deep Learning Challenging for Printed Circuit Board (PCB) Component Recognition and How Can We Address It?

Mukhil Azhagan Mallaiyan Sathiaselalan *, Olivia P. Paradis, Shayan Taheri and Navid Asadizanjani

Electrical and Computer Engineering Department, University of Florida, Gainesville, FL 32611, USA; paradiso@ufl.edu (O.P.P.); shayan.taheri@ufl.edu (S.T.); nasadi@ece.ufl.edu (N.A.)

* Correspondence: mukhil.mallaiyan@ufl.edu

Abstract: In this paper, we present the need for specialized artificial intelligence (AI) for counterfeit and defect detection of PCB components. Popular computer vision object detection techniques are not sufficient for such dense, low inter-class/high intra-class variation, and limited-data hardware assurance scenarios in which accuracy is paramount. Hence, we explored the limitations of existing object detection methodologies, such as region based convolutional neural networks (RCNNs) and single shot detectors (SSDs), and compared them with our proposed method, the electronic component localization and detection network (ECLAD-Net). The results indicate that, of the compared methods, ECLAD-Net demonstrated the highest performance, with a precision of 87.2% and a recall of 98.9%. Though ECLAD-Net demonstrated decent performance, there is still much progress and collaboration needed from the hardware assurance, computer vision, and deep learning communities for automated, accurate, and scalable PCB assurance.

Keywords: hardware assurance; PCB assurance; reverse engineering; bill of materials; AutoBoM; automated optical inspection; automatic visual inspection



Citation: Mallaiyan Sathiaselalan, M.A.; Paradis, O.P.; Taheri, S.; Asadizanjani, N. Why Is Deep Learning Challenging for Printed Circuit Board (PCB) Component Recognition and How Can We Address It? *Cryptography* **2021**, *5*, 9. <https://doi.org/10.3390/cryptography5010009>

Received: 19 January 2021

Accepted: 10 February 2021

Published: 1 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Printed circuit boards (PCBs) are essential components of many contemporary electronic systems, ranging from private sector computers and cell phones to public sector military and medical equipment. Predominant developments in the outsourcing of PCB manufacturing make boards increasingly vulnerable on a global scale to malicious changes from external entities [1]. Attacks such as hardware Trojan insertions provide backdoor access to sensitive networks and compromise a nation's infrastructure, military capability, and civil safety. Therefore, PCB assurance is an important research area in cyber-security.

With the PCB supply chain being potentially highly unreliable and insecure, the US government has taken multiple steps to bring changes to the protocols [2,3]. However, assurance is limited by the lack of automatic inspection methods. Reference [2] reports that apart from integrated circuits (ICs), resistors, capacitors, and transistors are among the most commonly counterfeited components, these being the most predominant of electronic components on any given PCB; it is, therefore, essential to identify and validate these components in any PCB.

Over the years, many state-of-the-art approaches for automated PCB assurance have been developed using image processing, computer vision, and machine learning in the visual spectrum [4–6]. However, most of these approaches often need a golden sample, i.e., a PCB proven free of flaws and hardware Trojans, to compare with the device under test (DUT). These golden references are often not available in areas such as hardware assurance reverse engineering. Hence, certain approaches that can operate in their absence must be developed, such as bill of materials (BoM) analysis.

A BoM is a list of all components on a PCB, such as resistors, capacitors, and integrated circuits (ICs) [7]. BoM analysis, i.e., contrasting the BoM collected from the DUT and the BoM published by the design firm, can reveal out-of-spec, defective, reused, recycled, and

malicious board alterations without the presence of a golden sample. BoMs have several uses outside the area of hardware assurance as well, such as reverse engineering (e.g., foreign, competitor, or legacy device analysis), industrial assessments (e.g., cost estimation, quality assurance), and academic research (e.g., technology trend analysis) [8]. Toward this end, we developed AutoBoM, which is a framework for automatically extracting a BoM using optical images of a target PCB [9].

For a successful and fully automated bill of materials extraction, several conditions must be met. Firstly, there are a various types of components on a board (resistors, capacitors, sensors, ports, potentiometers, inductors, etc.), so it is important to classify them into different types, and extract the peripheral information necessary to identify them correctly. For example, resistors can be identified by their resistance values and ICs can be identified by their model and part numbers. Second, processing should be performed extremely fast. Due to increased outsourcing and globalization, the electronics supply chain and hardware community span over multiple continents. With so many external entities involved in the PCB life cycle, companies must validate the PCBs quickly to maintain the short time-to-market. Finally, and most importantly, the automatic extraction of a bill of materials must be accurate. Though PCBs consist of hundreds, or even thousands of components, the addition, removal, or substitution of a single component could compromise the confidentiality, integrity, and/or accessibility of the entire system. An exemplary case is presented in Figure 1. Though the claims in [10] are under dispute, many experts in the hardware assurance community agree that the claims are highly plausible, and it is only a matter of time before more such reports surface. Since PCBs are often mass-produced for critical government, military, and biomedical infrastructures, 100% accuracy is highly desirable.

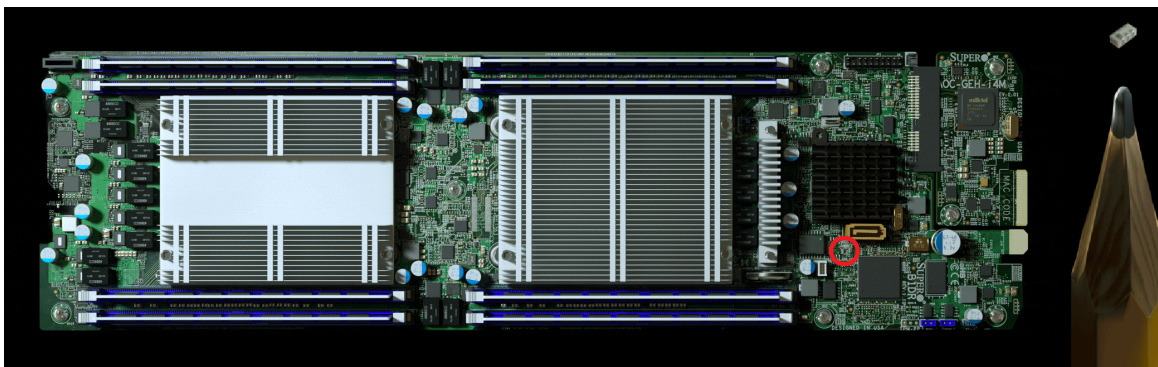


Figure 1. The Super Micro Computer, Inc. servo motherboard featured in Bloomberg Businessweek’s article, “The Big Hack.” Among the hundreds of components on the board, the suspected hardware trojan (circled in red) is no larger than a pencil tip, yet it compromised over 30 U.S. companies [10].

Numerous object detection approaches still cannot achieve near 100% accuracy on popular million-image datasets [11,12]. There is a trend of showcasing increasingly better performance either in speed or in accuracy over the other popular architectures, but the focus has not been on pushing toward attaining perfect accuracy for any one particular scenario [13,14], predominantly because this is not the focus. Automatic bill of materials extraction, being a special case of object detection, has not seen as a solution either even with innovative machine learning (ML) and deep learning (DL) techniques. Due to the inherent uniqueness of the PCB assurance field, such a solution is even more challenging and there is still much progress and collaboration needed from the hardware assurance and security community for automated, accurate, and scalable PCB component recognition.

Our contributions in this study are as follows: (i) an investigation of ML and DL challenges in the PCB assurance domain, (ii) an introduction of our proposed ECLAD-Net, a specialized network for PCB component detection, and (iii) a comparison of the benefits and limitations of the proposed method and those of other popular object detection methods. The rest of the paper is organized as follows. In Section 2, the challenges for

automated PCB assurance are elaborated. Then, in Sections 3 and 4, existing techniques along with a novel method, the electronic component localization and detection network (ECLAD-Net), are presented. Section 5 compares the proposed method with the existing methods. Finally, Section 6 concludes the paper with future work needed to enhance the hardware assurance community.

2. Challenges for Automatic PCB Assurance

There are several elements of PCBs that present challenges for component recognition and are hence difficult for existing algorithms to tackle. The following subsections elaborate on the challenges of (i) high intra-class/low inter-class variance, (ii) the evolving problem scope, and (iii) a limited amount of data.

2.1. The Variance Problem

PCBs are comprised of several different types of components. Examples of four common types are presented in Figure 2. As shown, there are examples of the same component type that appear different (high intra-class variance) and examples of different component types that appear similar (low inter-class variance). There are also many other types of components that further complicate the variance problem not shown in Figure 2, including but not limited to, inductors, transformers, and crystal oscillators. At present, there is a lack of universal standards for classifying the different component types (e.g., a resistor network array could be categorized as a resistor, an IC, or as a class all by itself).

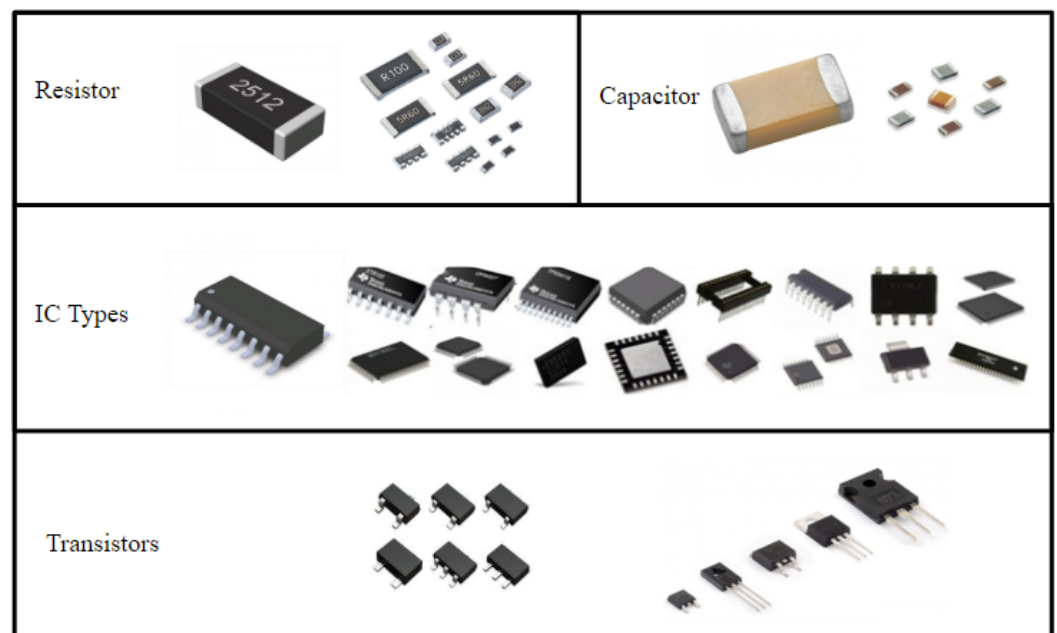


Figure 2. Examples of four common types of PCB components. High intra-class variance is shown in a variety of different integrated circuits (ICs), and low inter-class variance is shown in the similarities between some 3-pronged ICs and the transistors. Even resistors and capacitors, the most common components, can appear similar under certain lighting conditions.

In addition, there are numerous other factors that contribute to the variance problem. The PCB board itself may possess component-like features, which could result in false positive detections. Component appearances, functionalities, and classifications may vary by company, year, and even by designer. Components can also change over time, as they age naturally with standard wear-and-tear. Additionally, the majority of modern PCB components are machine-placed off-the-shelf surface mount device (SMD) resistors and capacitors. Such a class imbalance must be accounted for when training and evaluating component recognition algorithms. In addition, since the input image is an optical image,

there are also non-PCB factors contributing to the variance problem, such as imaging condition (e.g., lighting, angle-of-view, and resolution). These variations present additional challenges for designing a high-accuracy PCB component recognition method.

2.2. Evolving Problem Scope

Technology is in a constant state of development toward smaller components which enables a higher level of complexity with more compact designs at the PCB level, from small single-board microcontrollers to large motherboards and GPUs [10]). Additionally, just as PCB designs are becoming more advanced, so too are hardware Trojans and the protective countermeasures against them. Not only is there a variance challenge (as described in Section 2.1), but the aforementioned problem is regularly changing over time.

The constantly evolving nature of the PCB assurance problem and its scope presents challenges for traditional image processing and computer vision approaches, which operate best in closed systems [15]. Hence, intelligent algorithms capable of learning (i.e., ML and DL) are necessary to achieve and maintain high accuracy and avoid becoming outdated.

2.3. The Need for a Dataset

As stated in Section 2.2, the PCB assurance problem is highly dynamic, so ML and DL are vital. This requires a large and regularly updated dataset of images labeled by subject matter experts (SMEs). Though there are several existing datasets of PCB components, there are few that are publicly available [16]. Moreover, many of the available datasets are intended and formatted for a specific study, so they are difficult to combine effectively and re-purpose for a different study. For example, FICS-PCB [16] consists of 9900 images taken with a Digital Single-Lens Reflex (DSLR) camera and optical microscope with six component classes annotated, and PCB-Metal [17] consists of only 1000 images taken with a DSLR camera with four component classes annotated. While these datasets provide sufficient information to experiment with various algorithms, much work is needed to increase and maintain data to be representative of as many components and their variations as possible.

Meanwhile, PCB assurance is currently a field with a large number of classes and a relatively small amount of data per class. ML and DL methods, which tend to require large amounts of data, tend to overfit (i.e., overtrain) when trained on such a limited dataset. Overfitting, which is when a ML model is memorizing instead of learning salient features necessary for recognition, can be identified by significant performance degradation between training and testing accuracy. There are two primary ways to prevent overfitting: (1) collect more data or (2) use prior knowledge (e.g., PCB design rules) to reduce the number of free parameters so the amount of data required by the ML model is reduced. Both ways require much work and collaboration from the hardware assurance community to standardize the way data and prior knowledge is collected, annotated, and communicated.

To summarize, there are several object detection challenges in the PCB assurance domain. Hence, conventional approaches will be limited, as they are not tailored to the specific problem. In the following sections, we evaluate several of these conventional approaches, present a novel method, and compare them.

3. Related Work

Within the PCB assurance field, the majority of existing work has focused on defect detection. For example, [18] proposes a placement defect detection approach using genetic algorithms while [19] proposes an approach using SURF and conventional morphological operations. A comprehensive survey of old and new hardware defect detection approaches can be found in [1,4]. In such methods, “defects” are defined in reference to a golden, i.e., non-defective, sample. Counterfeiters and adversaries intentionally obscure designs to make detection difficult, but golden samples may not be always available. In addition, general analysis of a PCB to localize and extract regions using image segmentation techniques are presented in [20]. Similarly [21,22] present approaches using features based

detection (e.g., color, illumination, background, and tilted objects), however they address specific problems and not PCB assurance as a whole. Hence, alternative methods such as Automatic bill of materials extraction have been proposed as an additional hardware assurance method, complimentary to defect detection.

Automatic bill of material extraction is an application of object recognition, the process of detecting and identifying objects. Recognition can be categorized into two types: (i) regression and (ii) classification. The main difference is regression returns a continuous number (e.g., a confidence percentage such as an object is 78% capacitor vs. 22% resistor) while classification returns a discrete value (e.g., this object is most like a capacitor). Classification can be considered a special case of regression, in that a decision criteria is used to group the detected objects into preexisting categories (classes). In this study, the classification is considered for further analysis.

Over the years, numerous generic object recognition techniques have been proposed [23]. For example, popular two stage methods (e.g., RCNN [24], and FastRCNN [25]) and single stage detectors (e.g., YOLO [26] and SSD [27]) have all demonstrated high accuracy on popular datasets [23]. However, due to the inherent uniqueness of the PCB assurance field (as explored in Section 2), there is still much research is needed in this area.

Only recently have there been applications of ML and DL for PCB component recognition. Reference [28] proposes a conventional machine learning based approach using an AdaBoost classifier to detect capacitors. Here, the proposed method focuses only on through hole capacitors, which are easier to detect and are less common on modern PCBs than the SMD variety. Lately, there have been a few DL methods investigated for PCB component detection such as [29,30]. Reference [29] conducted a thorough analysis using the YOLO architecture and found Average Precision(AP) scores for capacitors and resistors at only about 50%. In [30], which uses an object detection with a graph-based neural network to recognize multiple PCB component classes, a similar poor performance is observed for smaller components such as resistors and capacitors. While both of the latter approaches appear to detect large components such as ICs, ports and those with high interclass variance, these struggle for smaller low interclass variance samples. From these works, it appears the challenges in PCB component detection are such that generalized techniques are not sufficient to achieve and maintain high accuracy. Hence, we propose a more specialized method, the electronic component localization and detection network (ECLAD-net) in the following section.

4. Materials and Methods

To reiterate, the goal of this paper is to present the need for specialized Artificial Intelligence (AI) for PCB hardware assurance. To this end, several generic object recognition methods from literature are explored. Then, we propose a novel specialized method: ECLAD-net, and compare its performance with those of the existing solutions.

Analysis is conducted in two phases: (i) classification and (ii) detection. Though the ordering of the phases appear counter-intuitive to traditional two-step detection-then-classification approaches, this is done on purpose to showcase the specific challenges associated with each step, independent of each other. In phase (i), we compare classifiers popular in literature with ECLAD-net's object classification stage using manually cropped resistors and capacitors. This experiment is conducted with such controlled conditions to investigate the feasibility of the tested classification methods, independent of any detection stages. In phase (ii), we compare object detection methods popular in literature with ECLAD-net's object detection stage using entire PCB images, with the goal to detect resistors and capacitors. Comparing only the detectors make it difficult to identify at what stage the error is from, thus making the classification phase necessary. Results are discussed in Section 5.

4.1. Experiment Specifications

The objective of the following experiments is to introduce and compare the performance of the novel ECLAD-net with those of popular approaches for the tasks of classifying and detecting resistors and capacitors. While there are other classes of components such as ICs, inductors; Resistors and capacitors succinctly capture the structure, color palette, and texture of the majority of components within a PCB. Results obtained using these two classes are sufficient to describe the complications in PCB component recognition.

Two datasets were used for the experiments in this study. For the component classification analysis, from 1152 images, 1000 images of manually cropped and labeled resistors and capacitors were used (to maintain class-balance). Data augmentation was used to increase the number of images to reduce overfitting of larger networks; more detail is discussed in the Sections 4.2 and 4.3. For the component detection analysis, 26 three-channel RGB images of PCBs were also used. The PCB images were collected with a high-resolution (i.e., 5000 pixels \times 4000 pixels, on average) DSLR camera, collected in a top-down fashion using a vertical mount setup. Table 1 presents statistics of the dataset used and Figure 3 presents examples of the datasets. Images were captured under moderate lighting conditions to reduce reflection from the different PCB materials. Image clarity is verified by computing the signal-to-noise ratio (SNR). PCBs are iteratively imaged until a clear, high resolution image, i.e., an image with a high db psnr, is obtained, relative to every sample imaged. Experiments were performed in python using libraries such as Keras, Opencv, Scikit-image, Scikit-learn, and Tensorflow. Where implementation of existing method is available, we have utilized them for comparison purposes. The code was run on a core i5 desktop with 4 GB of Nvidia 1050-Ti GPU.

Table 1. Image statistics for a database of resistors, capacitors, and full PCBs.

Image Name	Initial Dataset Images	Class-Balanced Dataset Images	Augmented Dataset Images	Average Size (in Pixels)	Resized Size (in Pixels)
Resistors	618	500	4000	29 \times 22	30 \times 30
Capacitors	534	500	4000	24 \times 41	30 \times 30
Full PCB	26	-	-	5729 \times 4113	-

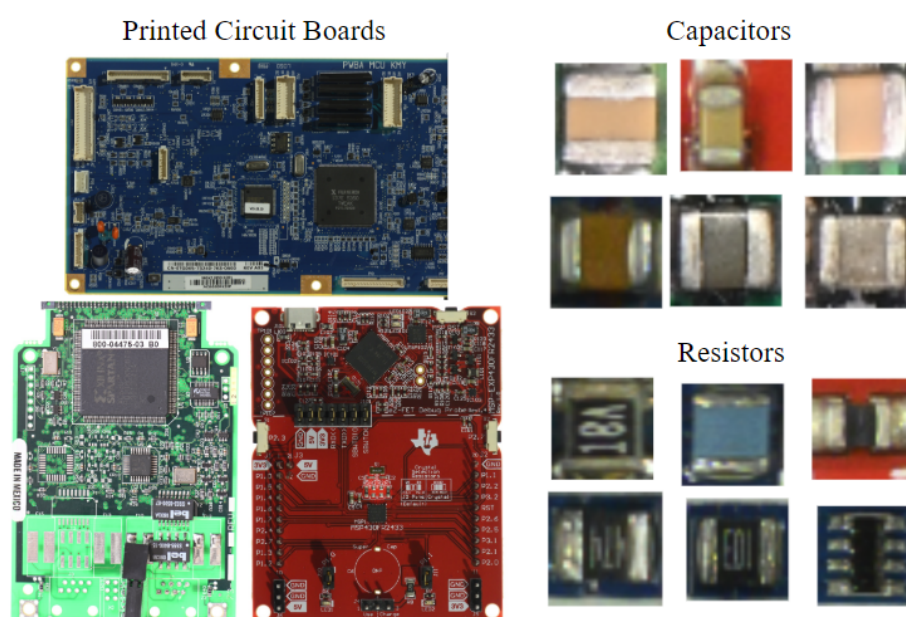


Figure 3. Sample data from the FICS-PCB dataset [16] used in this study. (left) Images of full PCBs used in the component detection analysis. (right) Labeled images of individual resistors and capacitors manually cropped from the full PCB images used in the component classification analysis.

4.2. Classification of Resistors and Capacitors

Before discussing the challenges of PCB component detection, classification of resistors and capacitors is first investigated to establish if the problem is feasible. Here, high intra-class/low inter-class variance is a key challenge. Hence, these classification experiments indicate class separability in the low- and high-dimensional spaces.

When training in higher-dimensional spaces, more data is required. Here, data augmentation was used to increase the dataset by a factor of 8. Augmentations were chosen to be realistic, i.e., representative of real-scenario variances. For instance, the capacitor and resistor images were rotated by 90° , 180° , and 270° . These values were chosen as realistic because pick-and-place assembly devices primarily implant SMD components horizontally or vertically with high precision and mild deviation. Additional augmentations include random image cropping with a maximum of 25% to encourage algorithm generalizability and regularization.

Additionally, before classification, images are normalized by scaling data to possess zero mean and unit variance. The dataset, which consisted of 8000 images after augmentation, was split 80%:20% into training and testing sets. The training data was further split 80%:20% for validation testing. Data was split to maintain balanced classes. When possible, 3-fold cross validation is performed. The loss function, optimization algorithms, and other algorithm-specific parameters vary by the method employed, and will be discussed in the following subsection.

4.2.1. Previous Classifiers from Literature

For a baseline, two conventional ML approaches were implemented: the squared error classifier and the hinge linear classifier. For both classifiers, two different models were used: one to model resistors, and the other for capacitors. Images were then classified based on the model that returned the highest confidence (i.e., match score). The hinge linear classifier models, which simulate linear support vector machines (SVMs) by optimizing over hinge loss, were configured as one vs rest. All linear classifier models were trained using stochastic gradient descent for 1500 epochs maximum, with a 10-epoch early stopping condition with an epsilon of 1×10^{-4} .

Several standard neural network (NN) approaches of various architectures were explored. Select examples are described below, including two fully-connected dense NNs, two convolutional NN (CNN), and VGG16 [31].

Two dense NNs were implemented: one with a single hidden layer (1024 nodes), and another with two hidden layers (1024 and 256 nodes). Each node employed a Rectified Linear Unit (ReLU) activation function [32]. Models were trained using an Mean Squared Error (MSE) cost and a Stochastic Gradient Descent (SGD) optimizer function with a step size of 0.0001. All the NN models were trained for a maximum of 150 epochs unless stopped early by the early stopping condition, similar to the ML classifiers.

Two CNNs were implemented: one with two convolution layers (16 and 32 filters), and another with three convolution layers (16, 32, and 64 filters). Convolution layers, which extract local image features, dramatically increase the number of trainable parameters relative to the amount of data. Hence, regularizing layers were utilized in the convolution layers: max pooling layers of size 2 and dropout layers with 25% probability. Here, only CNNs with two or three convolution layers are considered because (a) there isn't enough data to support a larger architecture and (b) at least 2 convolution layers are necessary to efficiently detect curves rather than just lines and summations of lines, such as the curved edges of many SMD capacitors (Figure 2). For both CNNs, the two hidden layer dense NN described in the preceding paragraph was incorporated after the feature-extracting layers. Models were trained using the same MSE cost function and an SGD classifier with step size of 0.0001, momentum of 0.5 and the same early stopping condition as before.

To explore deeper networks, a popular CNN architecture called VGG is also evaluated. Here, the VGG16 architecture, which has been the standard for many computer vision tasks, is programmed as in [31], with transfer learning done with our dataset.

4.3. Detection of Resistors and Capacitors

Detection of PCB components is the goal of this study. In the previous section (Section 4.2), existing classification methods were considered to gain insight in the class separability of the data in low- and high-dimensional spaces. In this section, existing methods are considered for detecting resistors and capacitors for comparison against our proposed method, the electronic component localization and detection network—ECLAD Net. Considering the various challenges outlined in Section 2, such as the limited amount of data in particular, we primarily consider simpler methods such as SSD [27] and RCNN [24].

For this experiment, 26 images of PCBs are used in addition to the 8000 images of resistors and capacitors described in Sections 4.1 and 4.2. Component images are used to re-train, through transfer learning, the previous object detection networks, which possess pre-trained weights. PCB images were then used to evaluate the object detection networks. Detection experiments follow similar train-validation-test splits, class balancing, and 3-fold cross validation as the component images in Section 4.2. Details on the implemented detection networks, along with any adaptations, are discussed in the following subsection.

4.3.1. Previous Detectors from Literature

Two single shot detectors (SSDs) were implemented: the SSD architecture as described in [27] and a modified version. The SSD proposed in [27], which builds off a VGG16 backbone, utilizes a pyramid of anchors to identify region proposals. The set of anchor boxes are manually selected based on component size. After the probabilities were determined, overlapping predictions are removed using Non-Maximal Suppression (NMS) [33]. For a simple baseline, a modified version of the SSD proposed in [27] is also implemented utilizing the CNN with three convolution layers described in Section 4.2.1 in place of the VGG16 backbone in the original version. SSDs were run on the PCB images in overlapping 300×300 -pixel windows. After NMS, results were merged as in [33] so resistors and capacitors were detected across the full PCB.

To explore two-stage object detection networks, a popular RCNN is also evaluated. In this study, the RCNN architecture is programmed as in [24]. Here, the RCNN, which uses an Alexnet [34] backbone, utilizes Selective search [35] to identify region proposals. Selective search follows the efficient graph-based segmentation method [36], which is designed to capture regions of interest at various scales with the trade-off of missing objects on the same scale. After the probabilities were determined, NMS and a linear SVM were used. The RCNN was run on the PCB images similar to the SSDs, but with overlapping 227×227 -pixel windows.

4.4. Electronic Component Localization and Detection Network

In this section, we propose the electronic component localization and detection network (ECLAD Net), a specialized neural network built for PCB component detection, presented in Figure 4. ECLAD-Net consists of two stages: region proposal and classification. Each stage is detailed in the following subsections.

4.4.1. Region Proposal Stage

As discussed in Section 2, the field of PCB assurance has many aspects of high intra-class/low inter-class variance which are unique. Unlike many conventionally-researched applications, which tend to feature natural scenes with a large variety of colors and textures, irregular contours, and sparsely-populated objects of interest, PCBs are more structured. For example, PCBs tend to feature artificial designs with a limited variety of colors, rectangular contours, and a multitude of densely-packed components with varying appearances. Since PCB backgrounds (i.e., the boards) tend to comprise a small distinct range of colors (e.g., green, blue, red), we developed a color-based background subtraction algorithm inspired by chroma-keying.

Though exact details are proprietary information and a patent has been applied for, the general procedure used in this study to achieve color-based background subtraction are as

follows. First, various-sized median and gaussian filters were used to reduce noise. Then, the original, unfiltered image and the filtered image were both converted into the Hue Saturation Value(HSV) color space. Since PCB backgrounds are relatively monochromatic and highly reflective, Hue channel thresholding and Saturation/Value reflection detection are used to model the background. The background model is then post-processed using binary morphological operations and used to isolate the individual PCB components in the unfiltered and filtered images. Region proposals are determined by averaging the two results. This is done to maintain both high frequency information (e.g., pins densely-packed resistors and capacitors) and low frequency information (e.g., packaging, ICs). A rectangle detector is then used to refine the region proposals, which may contain ICs, pins, ports, resistors, capacitors, etc. or just background. An example of the output of stage one is shown in Figure 5 (Left). Regions were then normalized and fed into the next stage of ECLAD Net.

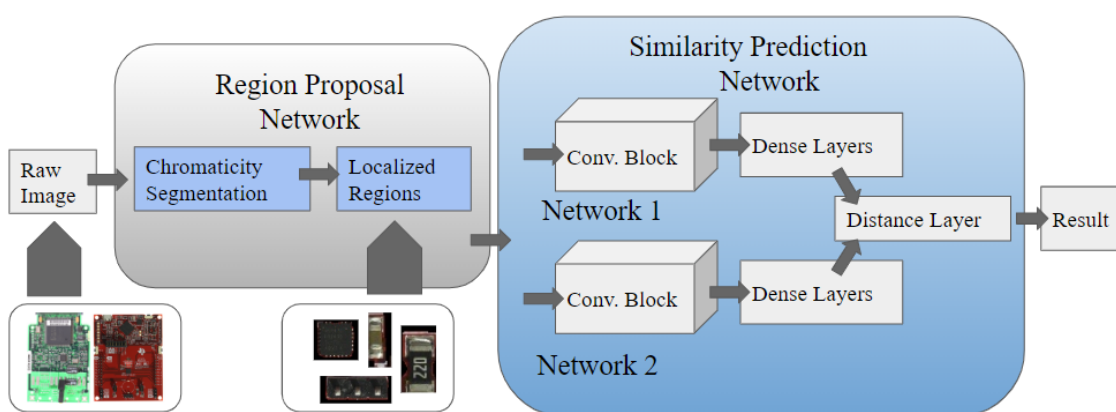


Figure 4. Proposed ECLAD-Net architecture for PCB component detection.

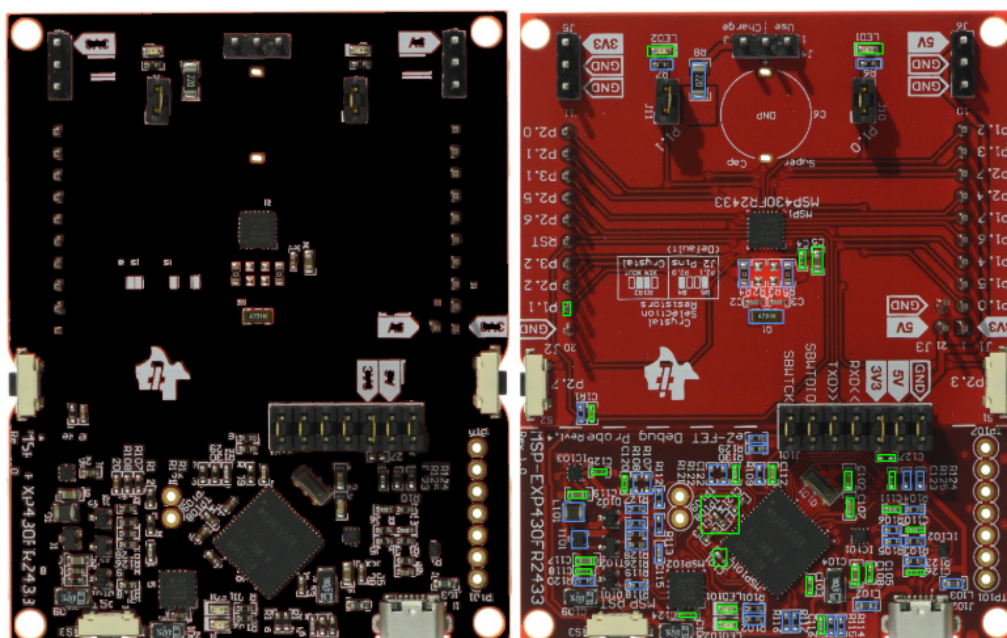


Figure 5. An example of ECLAD-Net results. (Left) Region proposals from the first stage. (Right) Resistors and capacitors detected from the classification stage, which are in blue and green boxes, respectively.

4.4.2. Classifier Stage

In addition to the region proposal stage, we also present a specialized classification stage: a similarity prediction network similar to [37]. In [37], two CNNs were trained

in parallel and the final layers were merged with a distance metric. This enables the network to learn both discriminative information. In this study, two modified 3 Layer CNN networks (as presented in Section 4.3.1) were used as the twin CNN architectures. The final layers were merged by applying the distance metric in Equation (1) below.

$$\delta(f(x_1), f(x_2)) = (f(x_1) - f(x_2))^T (f(x_1) - f(x_2)), \quad (1)$$

where $f(x_1)$ and $f(x_2)$ are the transformed input features from each of the twin CNN architectures. To capture the error for iterative learning, a weighted convex sum called contrastive loss [38] is used at the distance layer to ensure similar images are embedded closer in the feature space. Hence, minimization is done over the distance metric discussed in Equation (1) using the contrastive loss function in Equation (2).

$$L(x_0, x_1, y) = y\|f(x_0) - f(x_1)\| + (1 - y)\max(0, m - \|f(x_0) - f(x_1)\|), \quad (2)$$

where m , the margin parameter, was set to 1 in this study. As there are two classes in our study, we make use of two twin CNN setups for each class, i.e, two similarity prediction networks. During training, both twin CNNs were fed the same set of images and trained in parallel. However, one of the twin networks is specialized for classifying resistors, while the other is specialized for capacitors. During classification, for each twin network, one CNN is provided a template from the manually cropped dataset (a capacitor or resistor), and the other CNN iterates over the region proposal. Regions that are similar to the template image are then isolated with a match score; thus, for each region proposal we obtained a probability of it being a resistor and another probability of it being a capacitor. The class with the highest classification probability was deemed to be the predicted class. In addition, we performed the same test using three different templates for generalizability; the mean scores were then classified into the two classes. To extend this method to classify additional types of components, additional specialized networks are necessary for each class.

4.5. Evaluation

For the classification experiment, the classifier stage of ECLAD Net (Section 4.4.2) is compared with the existing classifiers described in Section 4.2.1. To evaluate the classifiers' performances, training and testing accuracy are used. A significant performance degradation from training to testing accuracy is indicative of model over-fitting. For the classification experiments, accuracy is a measure of recall given in Equation (3).

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \quad (3)$$

For further insight regarding the class separability of the data, t-SNE [39] was used. t-SNE is an unsupervised, non-linear feature reduction technique commonly used to visualize high-dimensional data by projecting it onto a low-dimensional manifold. Here, several t-SNE tests were conducted using various parameters; e.g., a perplexity of 10 to 30 and a step size of 10 to 20 were explored.

For the detection experiment, ECLAD Net (Section 4.4), is compared with the existing detectors described in Section 4.3.1. To evaluate the detectors' performances, in addition to recall, precision, given in Equation (4) is also used.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \quad (4)$$

An additional metric called F1-score is also evaluated. This is a harmonic mean between precision and recall scores and does not incorporate the biases of the individual metrics but rather combines their shortcomings into one, as seen in Equation (5).

$$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (5)$$

In the field of PCB assurance, a recall degradation indicates that trustworthy hardware is incorrectly flagged as suspicious, while a precision degradation indicates that suspicious hardware is incorrectly deemed trustworthy. Though the former may result in wasted resources (e.g., increased human SME involvement, tossed PCBs), the latter means an unsecured supply chain.

5. Results and Discussion

Results for the classification and detection experiments described in Section 4 are presented below.

5.1. Classification of Resistors and Capacitors

Table 2 presents the training and testing accuracy of the various classification methods, as described in Section 4.2. Our findings are detailed in the following paragraphs.

Table 2. The classification results for resistors and capacitors, with manually-cropped images given as input. The average accuracy from a set of five individual trials is reported.

Learning Model	Training Accuracy	Testing Accuracy
Squared Error Linear Classifier	51.6%	63.7%
Hinge Linear Classifier	95.2%	94.37%
1 Layer Dense NN	83.7%	82.8%
2 Layer Dense NN	96.1%	95.8%
2 Layer CNN	97.3%	97.2%
3 Layer CNN	98.9%	99.0%
VGG	98.1%	89.6%
ECLAD Net- Classifier Stage	99.1%	99.0%

The squared error linear classifier, one-hidden layer dense NN, and VGG16 demonstrated the lowest testing accuracy. The baseline squared error linear classifier performed as expected, as the classifier's assumption of a Gaussian distribution was not an accurate estimation of the data's Bernoulli distribution. Results of the one-hidden layer dense NN indicate that the data is too complex for such a small architecture to efficiently understand and model. On the other hand, the significant performance degradation between VGG16's training and testing accuracy implies over-fitting. This means there is not enough data for such a large architecture to regularize. Though the dataset can be augmented, there still remain limitations in the PCB assurance domain, especially concerning hardware Trojans (Section 2). These findings suggest that an ideal learning model should be nonlinear, larger than a one-hidden layer NN, and smaller than VGG16 in terms of the number and the size of layers.

The hinge linear classifier, two-hidden layer dense NN, and two-convolution layer CNN, all show accuracy in the 90 to 100 percentage range (Table 2). Results of the hinge linear classifier indicate that there exists a feature space in which the problem is linearly separable for the most part, but there still exists a few outliers. The two-hidden layer dense NN performed better than the one-hidden layer model, which supports that the latter model was indeed too small for this application. The results of the two-convolution layer CNN emphasize the importance of extracting local shapes and textures for detection and classification. These findings support that local images features are important for PCB component classification, as conceptually discussed in Section 2.

The three-convolution layer CNN and ECLAD-Net classifier consistently demonstrated the highest performance (Table 2). The three-convolution layer CNN performed slightly better than the two-convolution layer model, however both require hundreds of examples of each class. In contrast, the ECLAD-Net classifier can learn with as few as single-digit amounts of examples per class. This is ideal for the single-PCB scenario, where an operator or an end-user wishes to authenticate a purchased PCB. While resistors and capacitors are relatively common, there are many other types of components that are uncommon such as crystal oscillators; Zener diodes; and hardware Trojans, which are specifically designed to go undetected. Though 10 images per class is sufficient to obtain excellent results, the ECLAD-Net classifier's siamese network architecture [37] is inherently capable of training on a single instance of an image class with decent accuracy.

In addition to evaluating the training and testing accuracy for the classification experiments, a 3D t-SNE visualization of the resistor/capacitor separability is also trained and presented (Figure 6). Samples were collected to represent a wide variety of the possible resistors and capacitors. Though most instances were identified correctly, there were a few failed cases, which will be discussed later in Section 5.3.

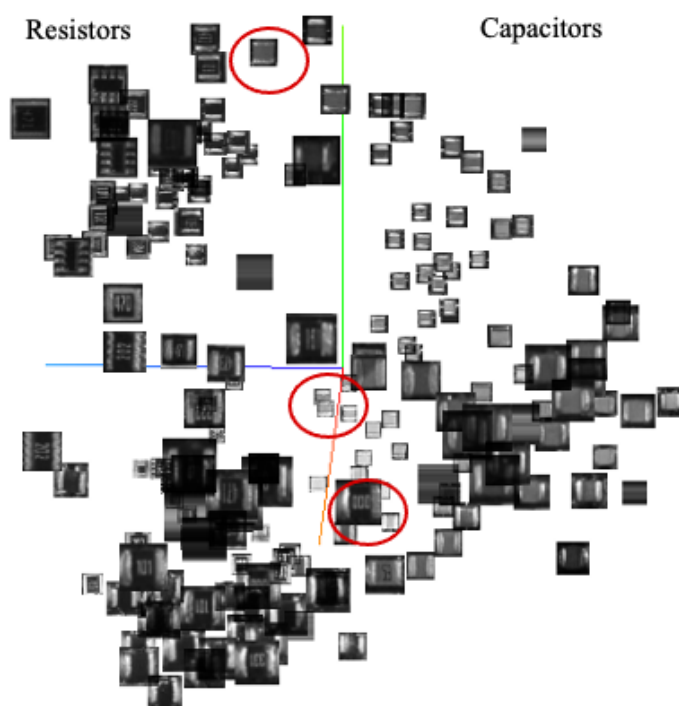


Figure 6. T-SNE visualization of resistor/capacitor class separability in 3D space [39]. Resistors are predominantly on the left side of the XZ plane (green and red), and capacitors are on the right. Images are in grayscale for visualization purposes only.

With the classification accuracy and the t-SNE visualization, we establish that resistors and capacitors, and in extension a variety of other PCB components, can be classified. In the next section, we discuss the results of the detection experiments.

5.2. Detection of Resistors and Capacitors

Table 3 presents the precision, recall, and F1 scores of the various detection methods, as described in Section 4.3. Findings are described in the following paragraphs.

As shown in Table 3, SSD, modified SSD, and RCNN demonstrated decently high recall scores, but relatively low precision. As discussed in Section 4.5, a low recall indicates trustworthy components are incorrectly flagged as suspicious, while a precision indicates suspicious components are incorrectly deemed trustworthy, a more dangerous scenario. In other words, though the three methods rarely miss resistors or capacitors, they return

numerous false positives. Such results are predominantly due to the number and quality of returned object proposals. Networks such as these are typically designed to locate objects with a high recall, while a threshold must be manually set to filter out false positives. However, as demonstrated, this generic approach is not always sufficient in the PCB assurance field, which exhibits very dense placements of low inter-class/high intra-class variance components (Section 2).

A notable object proposal complication, apart from missing objects, is the grouping of multiple components. In this study, SSD and RCNN had approximately half the proposed regions included multiple components, thereby complicating the predictions. Therefore, the accuracy of the region proposal system is just as important as the classification system. For example, SSD, which had a VGG backbone larger than the amount of data could support, returned more false region proposals and hence performed worse than the modified SSD, which had a smaller backbone. RCNN, with its selective search, faces similar problems. In one experiment, selective search returned more than 2000 region proposals, with only 75 instances of resistors or capacitors. Even after non-maximal suppression, many false positives still remained present. In the field, these large amounts of false positives, which must be manually-reviewed by human SMEs, are impractical for real-time PCB assurance. In addition to precision and recall, which describe only the effectiveness in a subset of the results, overall effectiveness of the approaches is described by the F1 scores in Table 3.

ECLAD-Net demonstrated the highest performance across all the evaluation metrics (Table 3). Its background subtraction technique, which is context-based yet still automatic, provides region proposals with a high accuracy in addition to a close-fitting contour such as the MaskRCNN [40]. In addition, ECLAD-Net's region proposal stage inherently prevents overlapped regions, which significantly reduces the number of false positive proposals and removes the necessity for non-maximal suppression. The number of region proposals returned by ECLAD-Net's first stage are on the magnitude of 100s, compared to SSD's anchor boxes and RCNN's selective search, which are on the magnitude of 1000s. ECLAD-Net's region proposal stage, which has only one parameter - a minimum area to capture, is able to operate at various scales.

In addition to the region proposal network, ECLAD-net's similarity-based classification network is also specialized to improve performance. This similarity-based network is effective for two reasons: (i) it learns discriminative information despite the low inter-class variance between resistors and capacitors, and (ii) it is capable of learning with few instances per class. In addition, ECLAD-Net can incorporate new information by either directly comparing the new samples with the old or by retraining on the newer samples and then comparing for a match.

Table 3. The detection results for resistors and capacitors, with PCB images given as input.

Learning Model	Recall Scores	Precision Scores	F1-Scores
SSD	88.4%	59.9%	71.41%
SSD- Modified	93.3%	67.4%	78.26%
RCNN	97.3%	53.7%	69.20%
ECLAD-Net	98.9%	87.2%	92.68%

5.3. Edge Cases

In this study, numerous challenges for PCB object detection were encountered. Though the proposed ECLAD-Net shows significant performance improvement compared to conventional approaches, completely autonomous PCB assurance is far from perfection. For instance, in the component segmentation stage, only about 90% of the components were captured in an unsupervised manner. Though it is possible to modify the parameters and produce near 100% accurate region proposals, such parameter tuning involves a lot of manual effort. Hence, there is still a need for a more intelligent segmentation algorithm.

Figure 6 shows examples of components that were misclassified in the classification experiments. As shown, the similar color and texture patterns make it difficult even for a human operator to identify. Some instances lie so close to the class separation boundary, they slightly overlap into the other region. Though the classification accuracies are above 99%, the field of PCB assurance demands near 100% accuracy (Section 1). Therefore, future research should focus on these challenges.

On the other hand, detection leaves a lot more room for improvement. Examples of edge cases are presented in Figure 7 and described in detail in the following sentences.

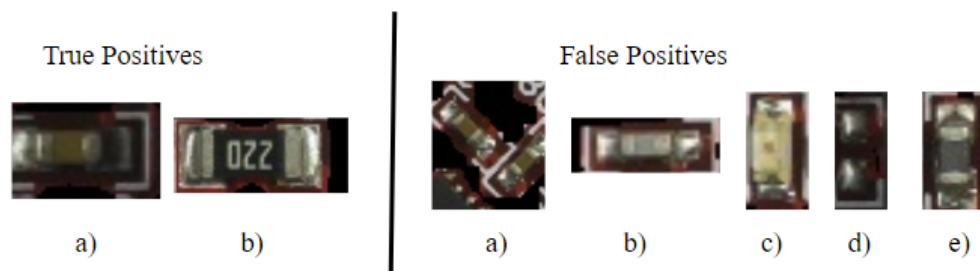


Figure 7. Challenging edge cases seen in the detection experiments. Shown on the left are true positive cases, i.e., difficult cases that were correctly recognized. On the right are false positive cases, i.e., difficult cases that were incorrectly recognized.

For the true positives, we see (a) a capacitor under non-ideal illumination due to being in the shadow of a large component, and (b) a large flat resistor with text in a dataset where the majority of resistors are small and without text. Despite these challenges, ECLAD-Net was still able to correctly recognize them. For the false positives, we see several examples. In Figure 7a, a horizontally-oriented capacitor is in the same region proposal as another. Here, ECLAD-Net correctly classified them as a capacitor, but failed to identify the two instances as separate. Figure 7b shows an uncommonly gray capacitor that was misclassified as a resistor. In Figure 7c, a light brown LED was misclassified as a capacitor due to its similarities to typical SMD capacitors. Figure 7d shows two solder pads under a shadow that were misclassified as a resistor. Though there is no component here, the shadow creates a false impression of a resistor. In Figure 7e, a dark gray inductor was misclassified as a resistor. For false positive cases, (c), (d), and (e), recognition can be difficult for even non-experts. Hence, as shown in Figure 7, there are several edge cases between resistors and capacitors, along with other types of components that were not trained on.

ECLAD-Net has shown to be effective in scenarios popular object detection networks fail, however the above edge-cases show that there is room for improvement by incorporating and learning multiple classes of components. Our current efforts in collecting and building a larger, more comprehensive and representative dataset will help this process and guide future research.

PCB authentication and assurance is another example of dense object detection for which there has been very limited research. It needs specialized approaches and increased attention from the computer vision community. An example of a similar problem is presented in spacenet [41] which works with dense satellite images; its objective is mainly to detect houses from satellite images, which makes it ultimately, a “one class” problem. Supermarket product detection [42], on the other hand, deals with multiple classes, but the accuracy of the predictions is not of high priority, yet identifying multiple classes is important. PCB component detection, however, needs accurate localization and classification of various microelectronic components, all of which share a high amount of variance in them. Thus, ECLAD-Net can be utilized for both of these purposes with slight modifications, as they are special cases of PCB component detection. Similarly, this is also applicable to problems in the science and medical spheres for the detection of small components in medical imaging, such as in [43].

6. Conclusions

In this paper, we proposed and described a network for PCB component detection to identify malicious, counterfeit, reused, or recycled components. We also presented a discussion of the unique challenges in PCB assurance, such as low inter-class/high intra-class variance, dense object placement, and limited data. Popular generic approaches for object detection, such as RCNN and SSD, do not provide the near-perfect performances desired in PCB assurance applications. Therefore, we present a more specialized network called the electronic component localization and detection network (ECLAD-Net). Results from various experiments were presented to compare the proposed ECLAD-Net with existing methods to showcase the need for such specialized networks.

Though ECLAD-net has demonstrated excellent classification and detection performance relative to the generic methods, there is still much room for improvement. While this paper focused on resistors and capacitors, similar problems arise with other components as well, such as ICs and transistors (Figure 2). Due to the uniqueness and importance of the PCB assurance field, attention and collaboration is necessary from both the hardware assurance community and the computer vision community. While this paper presents some fundamental research to help pave the way, there exists a larger scope for research and improvement, and we hope to continue research in this front.

In the future, we hope to improve our network to provide better precision and recall, while also improving algorithm speed. Component detection is only one part of PCB assurance. Text recognition for serial number verification, texture analysis for counterfeit detection, and 3D analysis for Trojan detection are just a few examples of other areas that would benefit from better and more efficient computer vision algorithms tailored toward PCB assurance and hardware security. In addition, there is also a great need for a larger and more comprehensive dataset for PCB assurance, as the lack of data is currently a bottleneck in applying DL to hardware assurance. To aid the community and help future research, we are in the process of building a public dataset and currently have around ten thousand component images.

Author Contributions: Conceptualization, N.A. and M.A.M.S.; methodology, M.A.M.S.; software, M.A.M.S.; validation, M.A.M.S.; formal analysis, M.A.M.S.; investigation, M.A.M.S.; resources, N.A.; data curation, M.A.M.S.; writing—original draft preparation, M.A.M.S. and O.P.P.; writing—review and editing, M.A.M.S., O.P.P., S.T., and N.A.; visualization, M.A.M.S.; supervision, N.A.; project administration, N.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The PCB Image dataset was released with the paper [16] and can be found at <https://www.trust-hub.org/#/data>. The manually-cropped dataset can be found at [Mukhil Mallaiyan, Sudarshan Agrawal, 2020. Resistor and Capacitor Image Dataset] https://github.com/mukhilazhagan/pcb_components_manual_crop; v 0.2.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
AP	Average Precision
BoM	Bill of Materials
CNN	Convolution Neural Network
DL	Deep Learning
DSLR	Digital Single-Lens Reflex
DUT	Device Under Test
ECLAD-Net	Electronic Component Localization and Detection Network
HSV	Hue Saturation Value
ICs	Integrated Circuits
ML	Machine Learning
MSE	Mean Squared Error
NMS	Non-Maximal Suppression
NN	Neural Network
PCB	Printed Circuit Board
RCNNs	Region Based Convolutional Neural Networks
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
SMD	Surface Mount Device
SMEs	Subject Matter Experts
SNR	Signal-to-Noise Ratio
SSDs	Single Shot Detectors
SVMs	Support Vector Machines

References

1. Mehta, D.; Lu, H.; Paradis, O.P.; Mallaiyan Sathiaselvan, M.A.; Rahman, M.T.; Iskander, Y.; Chawla, P.; Woodard, D.L.; Tehranipoor, M.; Asadizanjani, N. The Big Hack Explained: Detection and Prevention of PCB Supply Chain Implants. *ACM J. Emerg. Technol. Comput. Syst. (JETC)* **2020**, *16*. [\[CrossRef\]](#)
2. U.S. Department of Commerce. Defense Industrial Base Assessment: Counterfeit Electronics Report. Available online: <https://www.bis.doc.gov/index.php/documents/technology-evaluation/37-defense-industrial-base-assessment-of-counterfeit-electronics-2010/file> (accessed on 12 February 2021).
3. U.S. Government Accountability Office. DOD Needs to Improve Reporting and Oversight to Reduce Supply Chain Risk Report. Available online: <https://www.gao.gov/assets/680/675227.pdf> (accessed on 12 February 2021).
4. Moganti, M.; Ercal, F.; Dagli, C.H.; Tsunekawa, S. Automatic PCB Inspection Algorithms: A Survey. *Comput. Vis. Image Underst.* **1996**, *63*, 287–313. [\[CrossRef\]](#)
5. Richter, J.; Streitferdt, D.; Rozova, E. On the development of intelligent optical inspections. In Proceedings of the 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 9–11 January 2017; pp. 1–6.
6. Botero, U.J.; Wilson, R.; Lu, H.; Rahman, M.T.; Mallaiyan, M.A.; Ganji, F.; Asadizanjani, N.; Tehranipoor, M.M.; Woodard, D.L.; Forte, D. *Hardware Trust and Assurance through Reverse Engineering*; Association for Computing Machinery: New York, NY, USA, 2020.
7. Mcloughlin, I. Secure Embedded Systems: The Threat of Reverse Engineering. *Parallel Distrib. Syst. Int. Conf.* **2008**, 729–736. [\[CrossRef\]](#)
8. Quadir, S.E.; Chen, J.; Forte, D.; Asadi, N.; Shahbazmohamadi, S.; Wang, L.; Chandy, J.; Tehranipoor, M. A Survey on Chip to System Reverse Engineering. *J. Emerg. Technol. Comput. Syst.* **2016**, *13*, 6. [\[CrossRef\]](#)
9. Azhagan, M.; Mehta, D.; Lu, H.; Agrawal, S.; Chawla, P.; Tehranipoor, M.; Woodard, D.; Asadi, N. A Review on Automatic Bill of Material Generation and Visual Inspection on PCBs. In Proceedings of the ISTFA 2019: Conference Proceedings from the 45th International Symposium for Testing and Failure Analysis, ISTFA, Portland, OR, USA, 10–14 November 2019; pp. 256–265.
10. Jordan, R.; Riley, M. The Big Hack: How China Used a Tiny Chip to Infiltrate U.S. Companies., News Report, Bloomberg. 4 October 2018. Available online: <https://www.bloomberg.com/news/features/2018-10-04/the-big-hack-how-china-used-a-tiny-chip-to-infiltrate-america-s-top-companies> (accessed on 12 February 2021).
11. Everingham, M.; Eslami, S.M.A.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes Challenge: A Retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136. [\[CrossRef\]](#)
12. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255. [\[CrossRef\]](#)

13. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826. [CrossRef]
14. Zhao, Z.Q.; Zheng, P.; Xu, S.T.; Wu, X. Object Detection with Deep Learning: A Review. *IEEE Trans. Neural Networks Learn. Syst.* **2019**, *30*, 3212–3232. [CrossRef]
15. Botero, U.J.; Wilson, R.; Lu, H.; Rahman, M.T.; Mallaiyan, M.A.; Ganji, F.; Asadizanjani, N.; Tehranipoor, M.M.; Woodard, D.L.; Forte, D. Hardware Trust and Assurance through Reverse Engineering: A Survey and Outlook from Image Analysis and Machine Learning Perspectives. *arXiv* **2020**, arXiv:2002.04210.
16. Lu, H.; Mehta, D.; Paradis, O.; Asadizanjani, N.; Tehranipoor, M.; Woodard, D.L. FICS-PCB: A Multi-Modal Image Dataset for Automated Printed Circuit Board Visual Inspection. *IACR Cryptol. ePrint Arch.* **2020**, *2020*, 366.
17. Mahalingam, G.; Gay, K.M.; Ricanek, K. PCB-METAL: A PCB Image Dataset for Advanced Computer Vision Machine Learning Component Analysis. In Proceedings of the 16th International Conference on Machine Vision Applications (MVA), Tokyo, Japan, 27–31 May 2019; pp. 1–5. [CrossRef]
18. Xie, F.; Uitdenbogerd, A.; Song, A. Detecting PCB component placement defects by genetic programming. In Proceedings of the 2013 IEEE Congress on Evolutionary Computation, Cancun, Mexico, 20–23 June 2013; pp. 1138–1145. [CrossRef]
19. Hassanin, A.A.I.M.; Abd El-Samie, F.E.; El Banby, G.M. A real-time approach for automatic defect detection from PCBs based on SURF features and morphological operations. *Multimed. Tools Appl.* **2019**, *78*, 34437–34457. [CrossRef]
20. Kim, J.O.; Lee, Y.A.; Park, T.H. Automatic extraction of component inspection regions from Printed Circuit Board. In Proceedings of the 2012 IEEE/SICE International Symposium on System Integration (SII), Fukuoka, Japan, 16–18 December 2012; pp. 871–876.
21. Yuk, E.H.; Park, S.H.; Park, C.S.; Baek, J.G. Feature-learning-based printed circuit board inspection via speeded-up robust features and random forest. *Appl. Sci.* **2018**, *8*, 932. [CrossRef]
22. Lu, Z.; He, Q.; Xiang, X.; Liu, H. Defect detection of PCB based on Bayes feature fusion. *J. Eng.* **2018**, *2018*, 1741–1745. [CrossRef]
23. Jiao, L.; Zhang, F.; Liu, F.; Member, S.; Yang, S.; Li, L.; Feng, Z.; Qu, R. A Survey of Deep Learning-based Object Detection. *IEEE Access* **2019**, *7*, 128837–128868. [CrossRef]
24. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [CrossRef]
25. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
26. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
27. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. *SSD: Single Shot MultiBox Detector*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2015; Volume 9905. [CrossRef]
28. Fang, J.; Shang, L.; Gao, G.; Xiong, K.; Zhang, C. Capacitor Detection on PCB Using AdaBoost Classifier. *J. Phys. Conf. Ser.* **2020**, *1631*, 012185. [CrossRef]
29. Li, J.; Gu, J.; Huang, Z.; Wen, J. Application Research of Improved YOLO V3 Algorithm in PCB Electronic Component Detection. *Appl. Sci.* **2019**, *9*, 3750. [CrossRef]
30. Kuo, C.; Ashmore, J.D.; Huggins, D.; Kira, Z. Data-Efficient Graph Embedding Learning for PCB Component Detection. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 551–560. [CrossRef]
31. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks For Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556
32. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML'10), Omnipress, Madison, WI, USA, 4–8 July 2010; pp. 807–814.
33. Rothe, R.; Guillaumin, M.; Van Gool, L. *Non-Maximum Suppression for Object Detection by Passing Messages between Windows*; Lecture Notes in Computer Science; Cremers, D., Reid, I., Saito, H., Yang, M.H., Eds.; Springer: Cham, Switzerland, 2014; Volume 9003. [CrossRef]
34. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
35. Uijlings, J.; Sande, K.; Gevers, T.; Smeulders, A. Selective Search for Object Recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
36. Felzenszwalb, P.; Huttenlocher, D. Efficient Graph-Based Image Segmentation. *Int. J. Comput. Vis.* **2004**, *59*, 167–181. [CrossRef]
37. Koch, G.R. Siamese Neural Networks for One-Shot Image Recognition. 2015. Available online: <https://www.cs.cmu.edu/rsalakhu/papers/oneshot1.pdf> (accessed on 12 February 2021).
38. Hadsell, R.; Chopra, S.; Lecun, Y. Dimensionality Reduction by Learning an Invariant Mapping. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 17–22 June 2006; pp. 1735–1742. [CrossRef]
39. Maaten, L.v.d.; Hinton, G. Visualizing Data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.

40. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R.B. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2980–2988.
41. SpaceNet on, Amazon Web Services. “Datasets”. The SpaceNet Catalog. Last modified October 1st, 2018. Available online: <https://spacenet.ai/datasets/> (accessed on 12 February 2021).
42. Franco, A.; Davide Maltoni, S.P. Grocery product detection and recognition. *Expert Syst. Appl.* **2017**, *81*, 163–176. [[CrossRef](#)]
43. Manzo, M.; Pellino, S. Bucket of Deep Transfer Learning Features and Classification Models for Melanoma Detection. *J. Imaging* **2020**, *6*, 129. [[CrossRef](#)]